

THAT WHICH IS CLAIMED IS:

1. Method of handling branching instructions within a processor, the processor including a program memory containing program instructions, and a processor core (CR) containing several processing units (AU, DU) and a central unit (CU), in which the central unit, on receiving a program instruction, issues corresponding instructions to the various processing units, characterized in that, with the processor core (CR) being clocked by a clock signal, a branching instruction received by the central unit (CU) in the course of a current cycle is processed in the course of this current cycle.

2. Method according to claim 1, characterized in that a first processing unit (AU) contains at least one address-pointing register (Px), in that a branching instruction uses the content of at least one of the address-pointing registers, in that a check of the validity of the content of said pointing register in question is carried out at the start of said current cycle and in that said branching instruction is actually received by the central unit and processed if said content is declared valid, and, in the opposite case, this branching instruction is kept on hold for processing until said content is declared valid.

3. Method according to claim 2, characterized in that the content of each address-pointing register (Px) is recopied into a duplicated address-pointing register (PxC), and in that the check on the validity of the content of the pointing register in question is a check on the validity of the content of the corresponding duplicated register.

4. Method according to claim 3,
characterized in that, every time the central unit (CU)
receives a modifying instruction intended to modify the
content of an address-pointing register (Px), and
5 earlier in time than a branching instruction involving
this address-pointing register, the check of validity
of the content of the corresponding duplicated register
takes into account the fact that this modifying
instruction has or has not been processed by said first
10 unit (AU).

5. Method according to claim 4,
characterized in that, every time a modifying
instruction intended to modify the content of an
address-pointing register is received by the central
unit, a counter (CPTPx) associated with this register
5 is incremented, in that, every time this modifying
instruction has been processed by the addressing unit,
the counter is decremented, in that, when a branching
instruction involving this register is ready to be
10 issued to the central unit, said validity check
includes the check on the value of the counter, the
content of the duplicated register corresponding to
said address-pointing register involved being declared
valid if the value of the counter is equal to zero.

6. Method according to one of the preceding
claims, characterized in that a second processing unit
(DU) contains a guard-indication register (GR), in
that, in the presence of a guarded branching
5 instruction, a check on the validity of the value of
the guard indication assigned to said branching
instruction and contained in the guard-indication
register is carried out at the start of said current
cycle, and in that said guarded branching instruction
10 is actually received by the central unit and processed,
if the value of the corresponding guard indication is

declared valid, and, in the opposite case, this guarded branching instruction is kept on hold for processing until the value of the corresponding guard indication is declared valid.

7. Method according to claim 6, characterized in that the content of the guard-indication register is recopied into a duplicated guard-indication register (RGC), and in that the check on the validity of the value of a guard indication is a check on the validity of the value of the corresponding guard indication contained in the duplicated guard-indication register.

8. Method according to claim 7, characterized in that, every time the central unit receives a modifying instruction (GMix) intended to modify the value of a guard indication and earlier in time than a branching instruction guarded by said guard indication, the validity check on the value of the guard indication contained in the duplicated register takes into account the fact that this modifying instruction has or has not been processed by said second unit.

9. Method according to claim 8, characterized in that the processor core includes a memory of the FIFO type (DIDQ) associated with said second processing unit and intended temporarily to store the instructions which are intended for this second processing unit, in that, every time the central unit receives a modifying instruction (GMix) intended to modify the value of a guard indication, a counter, clocked by the clock signal, is initialized at an initial value corresponding to the number of clock cycles necessary for this modifying instruction to be stored in the memory (DIDQ), and in that said validity

15. stored value of the overflow bit of the write counter.

12. Processor, including a program memory containing program instructions, and a processor core (CR) containing several processing units and a central unit able, upon receipt of a program instruction, to
5 issue corresponding instructions to the various processing units, characterized in that, with the processor core being clocked by a clock signal, the central unit (CU) includes a branching module able to receive a branching instruction in the course of a
10 current clock cycle, and to process this branching instruction in the course of this current cycle.

13. Processor according to claim 12, characterized in that a first processing unit (AU) contains at least one address-pointing register (Px), in that a branching instruction uses the content of at
5 least one of the address-pointing registers, in that the central unit includes first validity-checking means (MCV1) able, at the start of said current cycle, to carry out a check on the validity of the content of said pointing register in question, and in that said
10 branching instruction is actually received by the central unit and processed if said content is declared valid, and, in the opposite case, this branching instruction is kept on hold in the program memory until said content is declared valid.

14. Processor according to claim 13, characterized in that the central unit includes, for each address-pointing register (Px), a duplicated address-pointing register (Px_C) the content of which is
5 a copy of the corresponding address-pointing register, and in that the first validity-checking means (MCV1) are able to check the validity of the contents of the corresponding duplicated register.

15. Processor according to claim 14,
characterized in that the central unit includes first
deriving means (MCT1) able, every time the central unit
receives a modifying instruction intended to modify the
5 content of an address-pointing register (Px), and
earlier in time than a branching instruction involving
this address-pointing register, to derive a first flag
signal (SGD1) representative of the fact that this
modifying instruction has or has not been processed by
10 said first unit.

16. Processor according to claim 15,
characterized in that the first deriving means (MCTP)
include:

- a counter (CPTPx) associated with each
5 duplicated address-pointing register (Px),
- incrementation means (MID) able, every time
a modifying instruction intended to modify the content
of an address-pointing register is received by the
central unit, to increment the counter associated with
10 this register,
- decrementation means (MID), able to
decrement the counter every time this modifying
instruction has been processed by the first processing
unit,
- 15 - comparison means (CMPx) able to compare the
value of the counter with the zero value and to issue
the first flag signal (SGD1) the value of which is
representative of the result of said comparison,
and in that the first validity-checking means (MCV1)
20 check the value of the first flag signal, the content
of the duplicated register corresponding to said
address-pointing register involved being declared valid
if the value of the flag signal corresponds to a value
of the counter equal to zero.

17. Processor according to one of claims 12 to 16, characterized in that a second processing unit (DU) contains a guard-indication register (GR), in that the central unit includes second validity-checking means (MCV2) able, in the presence of a guarded branching instruction, to carry out a check on the validity of the value of the guard indication assigned to said branching instruction and contained in the guard-indication register, at the start of said current cycle, and in that said guarded branching instruction is actually received by the central unit and processed if the value of the corresponding guard indication is declared valid, and, in the opposite case, this guarded branching instruction is kept on hold for processing until the value of the corresponding guard indication is declared valid.

18. Processor according to claim 17, characterized in that the central unit (CU) includes a duplicated guard-indication register (GRC) the content of which is a copy of the guard-indication register (GR), and in that the second validity-checking means are able to check the validity of the value of a guard indication contained in the duplicated guard-indication register.

19. Processor according to claim 18, characterized in that the central unit includes second deriving means (MCTG) able, every time the central unit receives a modifying instruction (GMIX) intended to modify the value of a guard indication (Gx) and earlier in time than a branching instruction guarded by said guard indication, to derive a second flag signal (DEGXV) representative of the fact that this modifying instruction has or has not been processed by said second unit.

20. Processor according to claim 19,
characterized in that the processor core includes a
memory (DIDQ) of the FIFO type associated with said
second processing unit and intended temporarily to
5 store the instructions which are intended for this
second processing unit, in that the second deriving
means include:

a counter (CPTx) clocked by the clock signal,
initialization means able, every time the central unit
10 receives a modifying instruction (GMIX) intended to
modify the value of a guard indication, to initialize
the counter at an initial value corresponding to the
number of clock cycles necessary for this modifying
instruction to be stored in the memory (DIDQ),

15 - logic means (MLG) receiving, on the one
hand, a first logic signal (SL1x) representative of the
current value of the counter and, on the other hand, a
second logic signal (DUGxV) representative:

20 - of the presence or the absence of the
modifying instruction in the memory, and
- of the fact that the guard-indication
register has or has not been updated by this modifying
instruction after the instruction has left said memory,
in that the logic means issue the second flag signal
25 (DEGxV),

and in that the second validity-checking
means (MCMV2) check the value of the second flag signal.

21. Processor according to claim 20,
characterized in that the second unit (DU) includes
control (MCTL) means able to determine the presence or
the absence of the modifying instruction in the memory,
5 in that these control means include:

a read counter incremented every time an
instruction is extracted from the second memory,

a write counter incremented every time an
instruction is stored in the second memory,

10 a set of individual registers (GTx)
associated respectively with the set of guard
indications,

 a first control unit (MC0) able, every time
an instruction modifying the value of a guard
15 indication is stored in the second memory, to store the
current value of the write counter in a field (CHx) of
the individual register associated with this guard
indication,

 a second control unit (MC1) able to determine
20 the still-present character of this modifying
instruction in the memory, and including means of
comparing said field of the individual register with
the current value of the read counter.

22. Processor according to claim 21,
characterized in that the write counter and the read
counter have an identical size equal to the depth of
the second memory, in that an overflow bit, changing
5 value every time the corresponding counter comes back
to its initial value, is associated with each counter,
in that each individual register further includes a
one-bit auxiliary field (Bax), in that the first
control unit (MC0) is able, every time an instruction
10 modifying the value of a guard indication is stored in
the second memory (DIDQ), also to store the current
value of the overflow bit of the write counter in the
auxiliary field (Bax) of the corresponding individual
register, in that the second control unit includes
15 auxiliary comparison means able to compare the current
value of the overflow bit of the read counter with the
content of the auxiliary field (Bax).

23. Processor according to claim 22,
characterized in that the auxiliary comparison means
include an EXCLUSIVE NOR logic gate (PLO1).

24. Processor according to one of claims 12 to 23, characterized in that it has a decoupled architecture.